

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 04-313121

(43)Date of publication of application : 05.11.1992

(51)Int.Cl.

G06F 9/38

(21)Application number : 03-236960

(71)Applicant : HITACHI LTD

(22)Date of filing : 23.08.1991

(72)Inventor : SUTETIBUN JII HAJI
TOORU BAJI

(30)Priority

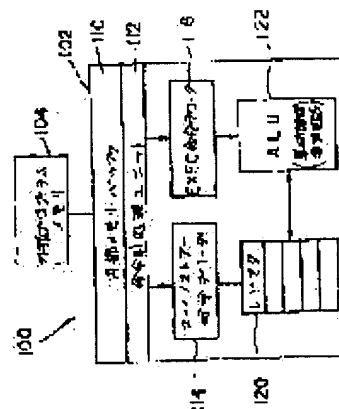
Priority number : 90 575140 Priority date : 29.08.1990 Priority country : US

(54) INSTRUCTION MEMORY DEVICE

(57)Abstract:

PURPOSE: To produce a long effective instruction word to the high speed processing while using an instruction memory for the short words.

CONSTITUTION: An internal memory buffer 110 provided against a data processing unit stores a series of instructions supplied from an external program memory 104. The two instructions which are sequentially adjacent to each other and stored in the buffer 110 receive the accesses in the instruction fetch cycles respectively. An instruction pre-processing unit 112 connected to the buffer 110 recomposes those two adjacent instructions into a single long instruction word when both instructions are coincidence with a fixed standard for combination. If the two instructions are not coincident with the fixed standard, a long instruction word is produced and therefore the first one of both instructions is combined with NQP instruction. Under such conditions, the second instruction receives an access again in the next instruction fetching cycle as the first one of two adjacent instructions.



LEGAL STATUS

[Date of request for examination]

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

(51) Int.Cl.⁵

G 0 6 F 9/38

識別記号

3 1 0 A 8725-5B

庁内整理番号

F I

技術表示箇所

審査請求 未請求 請求項の数6(全 11 頁)

(21) 出願番号 特願平3-236960

(22) 出願日 平成3年(1991)8月23日

(31) 優先権主張番号 5 7 5 1 4 0

(32) 優先日 1990年8月29日

(33) 優先権主張国 米国 (US)

(71) 出願人 000005108

株式会社日立製作所

東京都千代田区神田駿河台四丁目6番地

(72) 発明者 ステイブン ジイ ハイジ

アメリカ合衆国, カリフォルニア 94065,

レッドウッド シティ, アボセット ド

ライブ 570, アpartment 8111

(72) 発明者 トオル バジ

アメリカ合衆国, カリフォルニア 94010,

パーリングゲーム, モンテロ アベニュー

1303

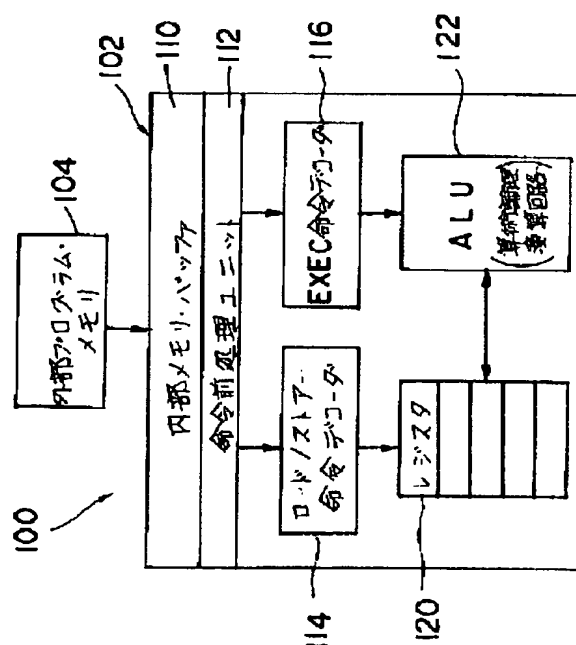
(74) 代理人 弁理士 玉村 静世

(54) 【発明の名称】 インストラクションメモリ装置

(57) 【要約】

【目的】 本発明の目的は、短い語のためのインストラクション・メモリを使用しながら、高速処理に対して有効な長い命令語を発生するための技術を提供することである。

【構成】 データ処理ユニットに対する内部メモリバッファ110は外部プログラムメモリ104から供給される一連の命令を記憶する。内部メモリバッファ110に記憶される2つのシーケンス状に隣接せる命令は、各命令フェッチ・サイクルにおいてアクセスされる。内部メモリバッファ110に接続されている命令前処理ユニット112は、2つの命令が組み合わせられるための既定の基準に一致するときに2つのシーケンス状に隣接せる命令を単一の長い命令語へ組み替える。既定の基準が一致しない場合には、長い命令語を発生するために、その2つの命令のうちの第1のものがノップ命令と組み合わせられる。かかる場合、その第2の命令は2つの隣接せる命令のうちの第1のものとして次の命令フェッチ・サイクル中に再びアクセスされる。



【特許請求の範囲】

【請求項1】 外部から取得した命令を保持して命令解読手段に与えるインストラクションメモリ装置であって、指定されたアドレス位置に第1の命令が記憶され、記憶された第1の命令を含む一対のシーケンス状に隣接する命令を同時にアクセスするための手段を含んでいて、一連の命令を一連のアドレス位置に記憶するための記憶手段と前記記憶手段に接続されていて、前記対にある命令が、組み合わせられるための既定の基準と一致するときに、前記対にある命令を単一の長い命令語へと組み替えるための命令前処理手段とを備えていることを特徴とするインストラクションメモリ装置。

【請求項2】 前記命令前処理手段は、既定の基準に一致しないときには、前記対にあるシーケンス状に隣接せる命令の前記第1の命令のみから長い命令語を発生することを特徴とする請求項1記載のインストラクションメモリ装置。

【請求項3】 前記記憶手段及び命令前処理手段に接続されていて、前記指定されたアドレスを発生するプログラム・カウンタを更に含み、前記プログラム・カウンタは、前記命令前処理手段が前記対にあるシーケンス状に隣接せる命令の前記第1の命令のみから長い命令語を発生するときに前記指定されたアドレスを1の値だけ漸増させ、前記命令前処理手段が前記対にあるシーケンス状に隣接せる命令を単一の長い命令語へと組み替えるときに前記指定されたアドレスを2の値だけ漸増させるための手段を備えていることを特徴とする請求項2記載のインストラクションメモリ装置。

【請求項4】 前記命令前処理手段は、前記既定の基準が適合されない場合に、前記対にあるシーケンス状に隣接する命令のうちの前記第1の命令のみから長い命令語を発生するために前記第1の命令を既定のノップ命令と組み合わせることにより当該長い命令語を発生する手段を含んでいることを特徴とする請求項2記載のインストラクションメモリ装置。

【請求項5】 レジスタ・ストア及びロード命令を解説する第1の命令デコーダと、演算手段で演算を行うための命令を解説する第2の命令デコーダとを含むデジタル信号処理ユニットのためのインストラクションメモリ装置であって、指定されたアドレス位置に第1の命令が記憶され、記憶された第1の命令を含む一対のシーケンス状に隣接する命令を同時にアクセスするための手段を含んでいて、一連の命令を一連のアドレス位置に記憶するための記憶手段と前記記憶手段に接続されていて、前記対にある命令が、組み合わせられるための既定の基準と一致するときに、前記対にある命令を単一の長い命令語へと組み替え、前記既定の基準に一致しないときには、前記対にあるシーケンス状に隣接せる命令の前記第1の命令のみから長い命令語を発生するための命令前処理手段とを備え前記既定の基準は、前記対にあるシーケンス

状に隣接せる命令の一方がレジスタ・ストア命令とレジスタ・ロード命令とから成るセットから選ばれた命令であり、そして前記対にある命令の他方が演算手段で演算を行うための命令であることを特徴とするインストラクションメモリ装置。

【請求項6】 前記記憶手段及び命令前処理手段に接続されていて、前記指定されたアドレスを発生するためのプログラム・カウンタを更に含み、前記プログラム・カウンタは、前記命令前処理手段が前記対にあるシーケンス状に隣接せる命令の前記第1のもののみから長い命令語を発生するときに前記指定されたアドレスを1の値だけ漸増させ、そして前記命令前処理手段が前記対にあるシーケンス状に隣接せる命令を単一の長い命令語へと組み合わせるときに前記指定されたアドレスを2の値だけ漸増させるための手段を含んでいることを特徴とする請求項5記載のインストラクションメモリ装置。

【発明の詳細な説明】

【0001】

【産業上の利用分野】 本発明は、外部から取得した命令を保持して命令解読手段に与えるインストラクションメモリ装置に関し、デジタル信号処理プロセッサに適用して有効な技術に関するものである。

【0002】

【従来の技術】 デジタル信号処理プロセッサは、フィルタリング、イコライゼーション、ノイズやエコーの除去、変調、フリーエ変換、そして画像の強調や特性パラメータの抽出などに利用される。このデジタル信号処理プロセッサなどのコンピュータ・アーキテクチャの中には、長さ60ビット又は64ビットのような長い命令語の使用が性能を効果的に高揚できるものもある。しかし、長い命令語の使用は、多くのプログラム命令がかかる長い命令語の使用を必要としないために、プログラムの記憶に関してシステムコストを上昇させる。尚、デジタル信号処理プロセッサについて記載された文献の例としては昭和59年11月30日株式会社オーム社発行の「LSIハンドブック」第593頁乃至第598頁がある。

【0003】

【発明が解決しようとする課題】 本発明は上記事情に鑑みてなされたものであり、短い語のためのインストラクション・メモリを使用しながら、高速処理に対して有効な長い命令語を発生するための技術を提供することを目的とする。

【0004】 本発明の前記並びにその他の目的と新規な特徴は本明細書の記述及び添付図面から明らかになるであろう。

【0005】

【課題を解決するための手段】 本願において開示される発明のうち代表的なものの概要を簡単に説明すれば下記の通りである。

【0006】すなわち、本発明はデータ処理ユニットに対するインストラクションメモリ装置を対象としている。内部インストラクション・メモリは一連の命令を記憶する。2つのシーケンス状に（連続的に若しくは所定の順番に従って）隣接する命令は各命令フェックサイクルにおいてアクセスされる。内部インストラクション・メモリに接続されている命令前処理ユニットは、2つのシーケンス状に隣接する命令が組み合わせられるための既定の基準と一致するときに、それら2つの命令を単一の長い命令語へと組み替える。2つの命令のうちの第1のものは、既定の基準が一致しないときに長い命令語を発生するために非演算命令つまりノップ命令と組み合わせられる。この場合、第2の命令は、2つの隣接する命令の第1のものとして次の命令フェック・サイクル中に再びアクセスされる。

【0007】ディジタル信号プロセッサなどにおいて、プログラム命令は一般に、プロセッサのレジスタへ値をロードするロード／ストア命令と、数学的計算を行う算術演算命令とに大別される。数学的計算はプロセッサのレジスタを利用しているが、本発明では、ロード／ストア命令が、同時に実行される算術演算命令により使用されつつあるレジスタに影響しない限り、ロード／ストア命令が算術演算命令と同時に実行されるという事実を利用する。

【0008】

【作用】本発明は、単一の命令への一連の短い命令語の組み替えが、セットにある既定の規則により決定されるような許容された組み合わせを含むときに、それら一連の短い命令語を外部プログラム・メモリに記憶するのを可能にする。このことがメモリ記憶の点で視する富むコストを低減させ、また、短い命令を長い命令へと動的に組み替えるのを可能にすることは、プログラムの実行速度向上若しくは処理能力向上に資する。

【0009】

【実施例】以下本発明に係る実施例を順次項目に従って説明する。

【0010】（好ましい実施例の説明）

【0011】図1には、外部プログラム・メモリ104に接続されたディジタル信号プロセッサ（DSP）102を持つデータ処理システム100が示されている。外部プログラム・メモリ104は、標準として、ランダム・アクセス・メモリ（RAM）又はリード・オンリ・メモリ（ROM）であるが、他の型式のメモリ・システムでも使用できる。いずれの場合においても、DSP102は、プログラム命令を外部プログラム・メモリ104からフェッチし、そしてそうした命令を、内部メモリ・キャッシュとして使用される内部メモリ・バッファ110に一時的に記憶する。例えば、内部メモリ・バッファ110は、各々の長さが32ビットである512命令語の記憶容量を持つことができる。データを内部メモリ・

バッファ110へとロードするめに使用される正確な機構及びタイミング制御は設置計画、つまりインプリメンテーションごとになる。

【0012】（短い及び長い命令語）

【0013】本発明において使用される基礎的原理によると、DSP102の内部では、命令セットにおける高度な直交性及び並行性の点から、長い命令語（例えば、60～64ビット）が使用される。しかし、32ビット・ワイド・プログラム・メモリ104及びDSP102と外部メモリとの間における32ビット・ワイド・インタフェースはそれらが64ビットワイド外部プログラムメモリに対するものよりもはるかに廉価なために、外部プログラムメモリ104は短い命令語（例えば、32ビット）を記憶する。命令前処理ユニット112は、対にある命令語を長い命令語に変換し、そしてそれらをDSPの命令デコーダ114及び116へ供給するために使用される。

【0014】DSP102は2つの命令デコーダ114及び116を持っている。第1の命令デコーダ114は、レジスタ120へ値をロードするために使用されるレジスタ・ロード及びストア命令を解釈するために使用される。第2の命令デコーダ116は、数学的計算を行うために算術論理演算回路122により実行される算術演算及び実行制御命令を解釈する。算術演算及び実行制御命令は、ここでは集成的に算術演算命令と呼ばれ、そうした命令を処理するためのハードウェア回路はEXECとして図示されている。

【0015】数学的計算はプロセッサのレジスタを利用しているが、並列命令デコーダ114、116の使用は、ロード／ストア命令が同時に実行される算術演算命令により使用されつつあるレジスタに影響しない限りロード／ストア命令が算術演算命令と同時に実行されるという利点を有する。従来のFIRフィルタ・プログラムはこの規則に従わないことに注意されたい。これは、内部乗算器が命令サイクルの始めにそれらレジスタのバージョンを使用すること、且つそれらレジスタに対して新しく計算された値がその計算において使用されることを意味するようである。レジスタの使用については、組み合わせでのLD／ST及びEXEC命令に関し、もしもあるならば、いかなる制約があるのかを特定することである。

【0016】図2には、短い命令を長い命令へと組み替える1つの方法が示されている。内部メモリ・バッファ（プログラム・バッファとも記す）110は2つの欄126及び128を持ち、欄126は算術演算命令（EXEC）を記憶するために使用され、欄128はロード／ストア命令（LD／ST）を記憶するのに使用される。外部プログラムメモリ104における各32ビット命令130は、1ビット命令識別子IID132と、31ビット命令コード134とを含んでいる。IIDは算術演

5

算命令に対して値0(ゼロ)に等しく、ロード/ストア命令に対して値1に等しい。

【0017】DSPのプログラム・カウンタ(以下単にPCとも記す)136が漸増するにつれて、命令はキャッシュ制御器138の制御の下で内部プログラム・バッファ110へと読み込まれる。各命令からのIIDビットにより制御される命令入力デマルチプレクサ140は、残りの31ビット命令コードを内部プログラム・バッファ110の算術演算命令欄126か又はロード/ストア欄128のいずれかへと切り換える。

【0018】もしもシーケンス状にフェッチされる二つの命令が或る既定の基準によって規定されるように適合可能であるとすると、両命令は内部メモリ・バッファ110の単一の欄へとそれぞれロードされ、それにより、それら2つの命令を単一の長い命令語へと組み合わせる。もしも2つのシーケンス状にフェッチされる命令が適合しなければ、そこでの命令は分離せる欄へとロードされ、そして、ノップ(つまり、NOP)命令は欄の1つの使用されていないスロットへとロードされる。

【0019】例えば、図3及び図4に示されている本発明の第1の好ましい実施例における既定の両立性つまり適合性規則は以下の通りである。つまり、もしも算術演算命令にロード/ストア命令が続くと、それら2つの算術演算命令のかたわらにロード/ストア命令を置くことにより単一の長い命令語へと組み合わせられる。もしも1つの実行命令が別の算術演算命令にすぐに続くとするならば、第1の算術演算命令に対する対応するロード/ストア・フィールドはノップで満たされ、そして第2の算術演算命令は内部プログラム・バッファの次の列に記憶される。同じことはシーケンスにおける2つ又はそれ以上のロード/ストア命令に対しても適用する。つまり、ノップは算術演算命令と組み合わせられないロード/ストア命令の算術演算命令フィールドに置かれることになる。

【0020】図5及び図6に示されている第2の好ましい実施例、そして図8及び図9に示される第3の好ましい実施例において、そこでの既定の適合性規則は第1の好ましい実施例におけるものと反対である。つまり、もしもロード/ストア命令に算術演算命令が続くとする、それら2つは、その算術演算命令のかたわらにロード/ストア命令を置くことにより単一の長い命令語へと組み替えられる。かくして、2つのシーケンス状にアクセスされる命令が長い命令語へ組み替えられるかどうかを決めるのに使用される規則としては多くの異なる規則がある。この発明の他の設置計画では、それらがプログラムにおいて生じる順序には無関係に、しかもそれら2つの命令がDSPでの異なるレジスタにアクセスするときのみ、算術演算命令をロード/ストア命令と組み合わせることができる。

【0021】プログラム命令を外部プログラム・メモリ

6

からじかにフェッチして実行するDSPはフェッチ・サイクル当たり1つの命令を実行するというような最大実行速度を持っている。本発明の内部メモリ・バッファを使用すると、プログラムがループか又は内部メモリ・バッファへと前もってフェッチされた他のコードを持つ場合、算術演算命令とロード/ストア命令とは同時に実行され、それにより、外部プログラム・メモリのみを使用するDSPの性能を2倍にしている。

【0022】同じ速度性能は、DSP上に62ビット又は64ビット・ワイド外部プログラム・メモリ及び同様に広いプログラム・メモリ・インタフェースを使用することにより達成される。しかしながら、かかる広いメモリ・インタフェースはコストの上昇を招き、しかも、算術演算命令及びロード/ストア命令が単一の長い命令語へと組み替えることのできなかったフィールドでは外部メモリに記憶される多数の命令がノップを含むために、極端に大きなプログラムが必要になる。

【0023】本発明を組み込んでいるDSPでもって最大のプロセッサ性能を達成するために、プログラマ、つまり、プログラム編集者は算術演算命令及びロード/ストア命令がインタリーブされるようにその命令シーケンスを計画しなければならない。かかるインタリーブは、算術演算命令及びロード/ストア命令が並列に実行されることを可能にする。もしもかかるインタリーブが使用できないとしても、プログラムは正確に実行され、しかもDSPのリソースもそれなりに利用される。

【0024】(第1の好ましい実施例)

【0025】図3及び図4を参照するに、そこにはそうした命令が適合可能であると思われるときにシーケンス状にアクセスされる命令を組み替えるための内部メモリ・バッファ110及び命令前処理ユニット150の1つのバージョンの好ましい実施例が示されている。この回路150は、一連の62ビット・ワイド並列命令を発生するように、直列にある32ビット・ワイド命令を2つの通路へと分割する。尚、図3と図4は①乃至⑤で示される対応箇所て相互に接続される。

【0026】一般に、図3及び図4に示されている回路150は、各フェッチ・サイクルをEXEC相とLD/ST相とに分割するように動作する。プログラム・カウンタ54が第1のクロック・サイクルのEXEC相の始めにインクリメントつまり漸増し、そして新しい命令がアクセスされるものと仮定する。もしもEXEC相中にアクセスされた命令が算術演算命令(IID=0)であるならば、プログラム・カウンタ154はLD/ST相の始めにおいてインクリメントされ、そして第2の命令はLD/ST相中にアクセスされる。もしも第2の命令が、IID=1(すなわち、第1のアクセスされた命令と組み合わせられる命令)を持つロード/ストア命令であるとする、その2つの命令はDSPの命令デコード14、116への伝達のためにパイプライン・レジスタ

186へと同時にロードされる。もしも第2の命令がロード/ストア命令でないとすると、ノップ命令はパイプライン・レジスタ186へと第1の(算術演算)命令と共にロードされ、そしてプログラム・カウンタ154は次のEXECクロック相の初めに漸増されない。もしも、EXECクロック相中にアクセスされる第1の命令がロード/ストア命令であったならば、それはノップ命令と共に、パイプライン・レジスタへとロードされ、そしてプログラム・カウンタ154は、次のEXECクロック相の始めまで、漸増されない。

【0027】更に特定するに、回路150は以下のように動作する。この実施例において、命令予備プロセッサ150はその命令フェッチ率を2倍において実行するためのライン152上におけるクロック信号を使用する。ライン152上におけるクロック信号は、そのクロック信号がANDゲート156により通過されるときに、プログラムカウンタ154を漸増つまりインクリメントする。

【0028】クロック信号は、2つの相補状のクロック信号、つまり、LD/ST(すなわち、ロード/ストア)相信号及びEXEC相信号をライン162及び164上にそれぞれ発生するために、フリップフロップ回路160により分割される。かくして、各命令フェッチ・サイクル(ライン152上におけるクロック信号の2つのクロック・サイクルからなる)は、ライン164上におけるEXEC相信号が可能化される(イネーブルレベル例えばハイレベルにされる)間のEXEC相と、ライン162上におけるLD/ST相クロック信号が可能化される間のLD/ST相とから成っている2つの相を持っている。

【0029】各アクセスされた命令からの命令識別子IIDはANDゲート166及び168によりそうした相クロックの各々と論理的にAND(論理積)される。もしも、アクセスされた命令が算術演算命令であることを意味するIID=0をそのアクセス命令が持つならば、ANDゲート166は、EXECクロック相中に、EXEC有効と呼ばれる可能化された出力信号を発生する。また、もしも、アクセスされた命令がロード/ストア命令であることを意味するIID=1をアクセス命令が持つならば、ANDゲート168は、ロード/ストア・クロック相中に、LD/ST有効と呼ばれる可能化された出力信号を発生する。2つの信号、つまり、EXEC有効及びLD/ST有効は、PCクロック・イネーブルと呼ばれる信号を発生するために、ORゲート170により論理的にオア(論理和)される。PCクロック・イネーブル信号がオンである場合にはいつでも、ライン152上におけるクロック信号の次のパルスがANDゲート156によって通過され、それにより、プログラム・カウンタPC154をインクリメントする。

【0030】ANDゲート166により発生されるEX

EC有効信号は、マルチプレクサ172が、ノップ命令か又はEXEC相中にキャッシュ・メモリ・バッファ110から読み出されたライン174上での命令コードのいづれをパイプライン・レジスタ176へロードするかを決定する。もしもEXEC相中にアクセスされた命令が算術演算命令であるならば、この命令はパイプライン・レジスタ176へとロードされるが、さもなければ、ノップ命令がロードされる。

【0031】同様にして、ANDゲート168により発生されるLD/ST有効信号は、マルチプレクサ180がノップ命令か又はLD/ST相中にアクセスされた命令のいづれをパイプライン・レジスタ182へロードするかを決定する。もしもLD/ST相中にアクセスされた命令がロード/ストア命令であるならば、この命令はパイプライン・レジスタ182へとロードされるが、さもなければ、ノップ命令がロードされる。

【0032】各フェッチ・サイクルの終り(すなわち、次のEXEC相の始め)において、パイプライン・レジスタ176及び182に記憶された命令コード及び/又はノップ命令は、それがDSP102のデコーダ114及び116により解説されるにつれて現行の命令語を記憶するために使用される別なパイプライン・レジスタ186へとロードされる。

【0033】最適の計算効率には交互せるロード/ストア命令及び算術演算命令でもってプログラムを順番に配列することによって達成され、もしも理想的な命令シーケンスが達成されないとしても、外部プログラム・メモリ104での空間は無駄にならない(内部キャッシュ・メモリ・バッファ110においても然りである)。

【0034】(第2の好ましい実施例)

【0035】この発明の第2の実施例を示す図5及び図6において、内部キャッシュ・メモリ200はそのキャッシュから2つの命令(すなわち、2列のデータ)を同時にアクセスするための二重アドレス・デコーダ202を持っている。内部キャッシュ・メモリの各列は、30ビット命令コード204と、2つのフラグ、つまり、命令識別子IID206に加えてグループ化フラグ208とを含んでいる。この実施例における命令識別子IID206は第1の好ましい実施例で使用されたものとは反対の定義を持っている。特に、長い命令の上位語及びロード/ストア命令に対するIIDは0に等しく、そして長い命令の下位語及び算術演算命令に対するIIDは1に等しい。かくして、この実施例は3つの型式の命令、すなわち、算術演算命令、ロード/ストア命令、及び“長い”命令(すなわち、本質的に、60~64ビットを占める命令)を想定している。尚、図5と図6は①乃至⑤で示される対応箇所相互に接続される。

【0036】この発明の実施例における命令前処理回路210は以下のように動作する。2つのシーケンス状に記憶される命令がキャッシュ200からアクセスされる

場合、もしも両命令からのグループ化フラグが等しいならば、そうした2つの命令は長い命令語へと組み合わせられる。もしもグループ化フラグが等しくないとする、ド位アドレス位置（ここでは、第1のアドレス位置と呼ばれる）からアクセスされる命令はノップ命令と組み合わせられ、この場合、第2のアクセスされる命令は次の命令フェッチ・サイクル中に（ジャンプ命令がプログラム・カウンタに新しい値を再びロードさせるのでなければ）使用される。かくして、この実施例において、EX EC及びLD/ST命令の順序は命令を組み合わせるかどうかを決定するのに使用されない。代って、各命令に対するグループ化フラグはこの決定をするのに使用される。

【0037】この実施例に従って実際に所望の論理を動作させるには、偶数フラグ(EF)信号を反転させる必要があるようで、PCが偶数アドレスを指しているとき、EF=1である。これをチェックしたほうがよい。

【0038】二つのグループ化されたフラグは、“異なるライン”信号DLを発生するために、排他的ORゲート211へ入力される。DLは、2つのグループ化フラグが等しくない（すなわち、長い命令語フォーマットを用いた場合、2つの命令はプログラムの異なるライン上にある）ならば、可能化（イネーブルレベルに）されるが、2つのグループ化フラグが等しい場合には無能化（ディスイネーブルレベルに）される。また、DL信号は、図5及び図6に示されているように、プログラム・カウンタ214が次の命令フェッチ・サイクルの始めに1又は2の計数だけインクリメントするかどうかを決定するのに使用される。プログラム・カウンタ214により出力されるアドレス値の最下位ビットは、偶数フラグ(EF)信号と呼ばれている信号を発生するために反転される。プログラム・カウンタが偶数アドレスにおける命令を指しているときでのEFは常に1である。スレーブ・ユニット215は、プログラム・カウンタの出力+1に等しい値を出力する。

【0039】2つの命令は各命令フェッチ・サイクル中に読まれるので、そうした2つの命令の命令識別子間を区別するために2つのラベルが使用される。特に、2つの命令はキャッシュ200でのシーケンシャル状態位置（所定の連続的な配置）から検索されるので、偶数アドレス値（すなわち、ゼロに等しい最下位ビットを持つアドレス）と、奇数アドレス値とを持つことになる。偶数アドレスにおける命令からのIIDはEIIDと呼ばれ、そして奇数アドレスにおける命令からのIIDはOIIDと呼ばれる。偶数アドレスにあるキャッシュの列からの命令コード204は母線つまりバス216上で伝送され、奇数アドレスにあるキャッシュの列からの命令コードはバス218上で伝送される。

【0040】次に、2つのアクセスされた命令は1つ又は両命令をパイプライン・レジスタ220へと適切にロ

ードするように前処理されなければならない。アクセスされた偶数列からのバス216上における命令コードは、もしもそれがロード/ストア命令(EIID=0)であるならば、バッファ224により“0”入力マルチプレクサ222へと通過される。もしもそれがプログラム・カウンタにより指定される命令であるならば、アクセスされた命令は共に、パイプライン・レジスタ220へとロードされることになる。バッファ224及び228に対する制御ロジック226は、もしもアクセスされた命令が共にロード/ストア命令であるとする、2つのロード/ストア命令が同時にライン230上に現われるのを回避させるように動作する。同様に、アクセスされた奇数列からのバス218上における命令コードは、もしもそれがロード/ストア命令(OIID=0)であるならば、バッファ228によりマルチプレクサ222の“0”入力へと通過され、そしてロジック226は、これがそのプログラム・カウンタにより指示されている命令であるのかあるいは両アクセスされた命令がパイプライン・レジスタ220へとロードされるべきなのかどうかを決定する。

【0041】バッファ232及び234は、マルチプレクサ222に対するのと同じ制御方法を用いて、ライン240を介してロジック238の制御の下で偶数及び奇数命令バス216及び218からの命令をマルチプレクサ236の“0”入力へと通過させるけれども、算術演算命令(EIID=1又はOIID=1)はそのマルチプレクサへと通過される。制御論理回路（ロジック）226及び238に対する図6に示されているブール論理式は一例であり、少ない数のインバータ、NANDゲート及びNORゲートを用いて容易に履行される。

【0042】制御論理回路250及び252は、マルチプレクサ222及び236がライン230及び240上での命令か又はノップ命令のいずれをパイプライン・レジスタ220へと通過させるのかを決定する。もしも2つの検索された命令が、DLを0に等しくする同じグループ化フラグを持つならば、論理回路250及び252は共に、“0”の値を出力して、両マルチプレクサに作用し、検索された命令（ライン230及び240上で見出せる）をパイプライン・レジスタ220へと通過させる。こうした命令は次の命令フェッチ・サイクルの始めにおいてそのクロック信号によりパイプライン・レジスタ220へとラッチされる。

【0043】もしも2つの検索された命令が、DLを1に等しくする、同じグループ化フラグを持たないならば、論理回路250は、プログラム・カウンタにより指示される命令が算術演算命令(OIID又はEIID=1)であることを条件にして、ノップ(NOP)をパイプライン・レジスタ220の“LD/ST”部分へ伝送し、さもなければ、その“0”入力ポート上における検索された命令をパイプライン・レジスタ220のLD/

ST部分へと通過させることになる。同様に、もしも2つの検索された命令が同じグループ化フラグ(DL=1)を持たないならば、論理回路252は、プログラム・カウンタにより指示されている命令がロード/ストア命令(OIID又はEIID=0)であることを条件にして、ノップ(NOP)をパイプライン・レジスタ220のEXEC部分へ伝送し、さもなければ、その"0"入力ポート上における検索された命令をパイプライン・レジスタ220へと通過させることになる。

【0044】図7を参照するに、そこには、図5及び図6に示されているこの発明の実施例に関連して使用するための命令メモリ・キャッシュ200の1つの実施例が示されている。このメモリ・キャッシュは2つのメモリ・バンク260及び262を持ち、バンク260は命令を偶数アドレス位置に記憶するために使用され、バンク262は命令を奇数アドレス位置に記憶するために使用される。そこにはまた、2つのプログラム・カウンタ、つまり、マスタ・プログラム・カウンタ264と、そしてカウンタ264の出力よりも1だけ大きい値を常に出力するスレーブ・プログラム・カウンタ266とがある。

【0045】両プログラム・カウンタには初期値がロードされている。マスタ・プログラム・カウンタには特定の値がロードされ、そしてスレーブ・プログラム・カウンタはそれをカウンタ回路270へとロードする前に1だけその値を漸増させるのにインクリメント回路268を使用する。その後、両カウンタ264及び266は、

(INSTRタイプ) (コード)

[コードの管理区間]

EXEC	R3=スケール	/*スケール定数中の負荷
EXEC	A0=\$0	/*データのためのベースアドレス
EXEC	A8=\$100	/*係数のためのベースアドレス
EXEC	MR=タップ_ナンバー	/*モジュロサイズ<フィルタタップナンバー
EXEC	A1=タップ_ナンバー2	/*計数を繰り返す
	fstart	
EXEC	待ち(SIR)	/*直列入力を待つ
EXEC	R4=sir	/*直列入力をフェッチ
EXEC	R4=フロート(R3)	/*データ変換
LD/ST	*A0=R4	/*最新データとして記憶
LD/ST	R0=*A0++% R1=*A8++%	
LD/ST EXEC	R0=*A0++% R1=*A8++%	
EXEC	A1を反復	

[コードのナンバー・クランピング区間]

```
LD/ST|EXEC R0=*A0++% /*フィルタ・コア
            R1=*A8++%
            ||FR2=R0*r1||FR5+=R2
)
```

[コードの管理区間]

```
EXEC A0--=$1 /*データ・ポインタ dec
```

新しい出発アドレスがプログラム・カウンタへとロードされるまで、各フェッチ・サイクルの始めにおいて同じ量(1又は2)だけ漸増される。

【0046】2つのプログラム・カウンタ264及び266からの出力は、どのカウンタ出力が各メモリ・バンク260、262へ行くべきかを決定するのに2つのマルチプレクサ272及び274を必要とするようになっている。マルチプレクサ272は2つの計数値のうち1つをメモリ・バンク260へ送出し、マルチプレクサ274は2つの計数値のうちの他の1つをメモリ・バンク262へ送出す。特に、マスタ・プログラム・カウンタ264により出力される計数値の最下位ビット(LSB)は各マルチプレクサ272、274のうちのどの入力を使用されるべきなのかを特定するのに使用される。もしもLSBが0に等しいならば、マスタ・プログラム・カウンタ264は偶数値を出力し(そしてスレーブ・カウンタ266は奇数値を出力する)、それ故、マスタ・プログラム計数値は偶数メモリ・バンク260のために使用され、そしてスレーブ・プログラム計数値は奇数メモリ・バンク262のために使用される。もしもLSBが1に等しいとすると、マスタ・プログラム計数値は奇数メモリ・バンク262のために使用され、そしてスレーブ・プログラム計数値は偶数メモリ・バンク260のために使用される。

【0047】本発明を適用するFIRフィルタ・プログラム(FIR DSP プログラム)の例を表記すると以下ようになる。

(内容/注釈)

13

EXEC R5=FI×(R3) /*データ変換
 EXEC SOR=R5 /*データ・アウト
 EXEC 待(SOR) /*直列出力を待つ
 EXEC fstartに戻る

【0048】上記プログラムにおける第1の管理区間に
 関して、EXEC及びLD/ST命令に対して30ビッ
 ト又は32ビットの長さの命令を使用すると、単一の3
 2ビット命令セットのみを使用するDSPにおけるより
 も一層の機能及びメモリ/レジスタ・リソースを可能に
 する。そのプログラムの“ナンバー・クランチング区
 間”は並列EXEC及びLD/ST演算を利用する。ナ
 ンバー・クランチング区間は短いので、すべての命令
 は、外部プログラム・メモリをアクセスすることなし
 に、メモリ・キャッシュ200からアクセスされること
 になる。更に、ナンバー・クランチング区間は短い
 が、それは多くの回数繰り返されて、かかる並列処理が信号
 処理プログラムの小さな数のラインのみににおいて使用さ
 れる場合でさえ、算術演算命令及びロード/ストア命令
 の並列処理を有効にする。

【0049】(第3の好ましい実施例)

【0050】図8及び図9に示されている実施例は、図
 5及び図6に示され且つ前に述べた第2の実施例の変形
 例である。特に、命令メモリ・キャッシュ300は1ビ*

$$/DL=(EF \cdot /EIID \cdot OIID) + (/EF \cdot EIID \cdot /OIID)$$

さもなければ、この実施例におけるハードウェアは、図
 5及び図6に示されている第2の好ましい実施例に対す
 るものと同じである。尚、上記ブール式において信号の
 前に付された記号/は反転レベルを意味する。

【0052】この実施例を使用して、もしもプログラマ
 が次のプログラム・シーケンス、つまり：

LD/ST ノップ
 ノップ EXEC

を意図的にプログラムしたいならば、プログラマは、外
 部メモリにおけるノップ命令をLD/ST命令とEXEC
 命令との間に挿入しなければならない。これは、最終
 的に実行されるプログラムに対する外部的に記憶される
 プログラムの圧縮比における幾らかの損失を意味する
 が、ノップ命令の明確なプログラミングを必要とする、
 かかる命令シーケンスはごくまれに生じるようである。

【0053】以上本発明者によってなされた発明を実施
 例に基づいて具体的に説明したが、本発明はそれに限定
 されるものではなく、その要旨を逸脱しない範囲におい
 て種々変更可能であることは言うまでもない。例えば外
 部プログラムメモリから供給される命令は32ビットに
 限定されず、16ビット等その他適宜のビット数であっ
 てもよい。

【0054】以上の説明では主として本発明者によって
 なされた発明をその背景となった利用分野であるディジ
 タル信号処理プロセッサに適用した場合について説明し
 たが本発明はそれに限定されるものではなくその他各種

14

*ット命令識別子IID302を各列に対して含むが、グ
 ループ化フラグを含んでいない。これは、各命令に対す
 る命令コードを、30ビットのみに代わって、31ビッ
 トの長さにするのを可能にする。

【0051】この実施例に対する命令プロセッサ回路3
 10を履行するには、偶数列命令識別子EIID、奇数
 列命令識別子OIID及び偶数フラグEFに基づいて“
 差ライン”DL信号(図においてオーバーバーが付され
 た信号であり、以下反転DL信号とも記す)を発生する
 特別な論理回路312が必要である。特に、反転DL信
 号が0に等しくそして2つの検索された命令は、第1の
 検索された命令がIID=0としたLD/ST命令であ
 りそして第2の検索された命令がIID=1としたEX
 EC命令であることを条件に長い命令語へと組み替えら
 れる。EF信号は、第1の検索された命令が偶数命令か
 又は奇数命令のいずれに対応しているのかを決めるため
 に論理回路312により使用され、それにより、次のブ
 ール公式が前記反転DL信号に対して使用されることを
 可能にする。すなわち：

のデータ処理ユニットに適用することができる。

【0055】本発明は、少なくとも外部から取得する命
 令語をそれよりもビット数の多い命令に組み替えて命令
 解読手段に供給する条件の者に適用することができる。

【0056】

【発明の効果】本願において開示される発明のうち代表
 的なものによって得られる効果を簡単に説明すれば下記
 の通りである。

【0057】すなわち、単一の命令への一連の短い命令
 語の組み替えが、セットにある既定の規則により決定さ
 れるような許容された組み合わせを含むときに、それら
 一連の短い命令語を外部プログラム・メモリに記憶する
 のを可能にするので、メモリ記憶の点ではシステムコス
 トの上昇を抑えることができ、且つ、短い命令を長い命
 令へと動的に組み替えるのを可能にすることによってプ
 ログラムの実行速度を向上させることができるという効
 果を得る。

【図面の簡単な説明】

【図1】図1は内部インストラクション・メモリと命令
 前処理ユニットを持つデジタル信号プロセッサのブロ
 ック図である。

【図2】図2は長い命令語を発生して記憶するための内
 部インストラクション・メモリバッファの概念的ブロッ
 ク図である。

【図3】図3は図4と共に示す本発明の第1の好ましい
 実施例のブロック図である。

15

16

【図4】図4は図3と共に示す本発明の第1の好ましい実施例のブロック図である。

【図5】図5は図6と共に示す本発明の第2の好ましい実施例のブロック図である。

【図6】図6は図5と共に示す本発明の第2の好ましい実施例のブロック図である。

【図7】図7は本発明のメモリバッファ部分を履行するのに適しているメモリバッファのブロック図である。

【図8】図8は図9と共に示す本発明の第3の好ましい実施例のブロック図である。

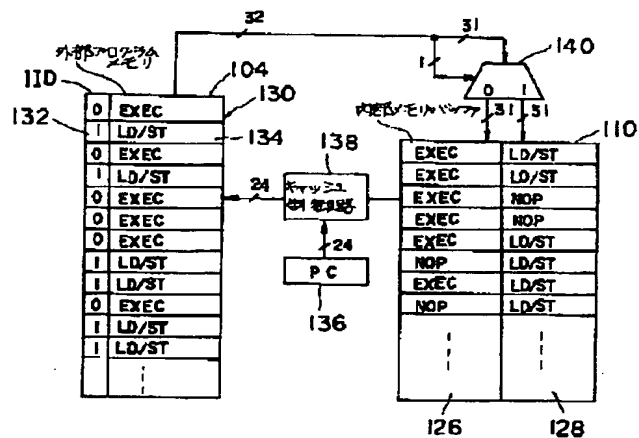
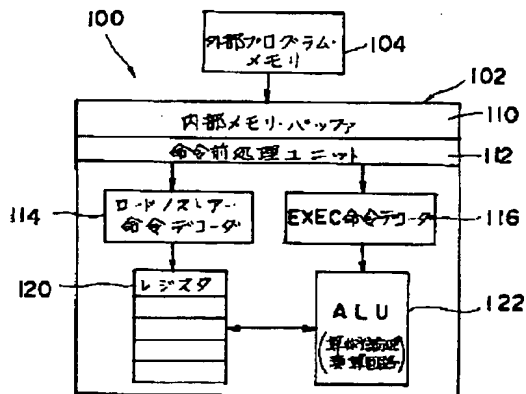
【図9】図9は図8と共に示す本発明の第3の好ましい実施例のブロック図である。

【符号の説明】

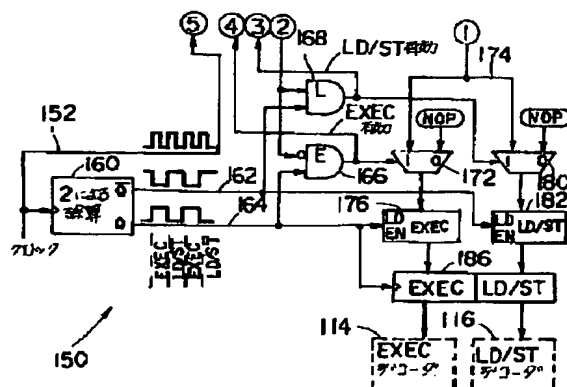
- | | | | |
|-----|----------------|----------|-----------------|
| 100 | データ処理システム | 136 | プログラム・カウンタ |
| 102 | ディジタル信号処理プロセッサ | 138 | キャッシュ制御器 |
| 104 | 外部プログラム・メモリ | 140 | 命令入力デマルチプレクサ |
| 110 | 内部メモリ・バッファ | 150 | 命令前処理ユニット |
| 112 | 命令前処理ユニット | 154 | プログラムカウンタ |
| 114 | ロード/ストア命令デコーダ | 160 | フリップフロップ回路 |
| 116 | EXEC命令デコーダ | 176 | パイプラインレジスタ |
| 120 | レジスタ | 182 | パイプラインレジスタ |
| 126 | 算術演算命令欄 | 186 | パイプラインレジスタ |
| 128 | ロード/ストア欄 | 200 | 内部キャッシュ・メモリ |
| 122 | 算術論理演算回路 | 202 | 2重アドレスデコーダ |
| | | 204 | 命令コード |
| | | 206 | 命令識別子 I I D |
| | | 208 | グループ化フラグ |
| | | 210 | 命令前処理ユニット |
| | | 214 | プログラム・カウンタ |
| | | 220 | パイプライン・レジスタ |
| | | 226, 238 | ロジック |
| | | 260, 262 | メモリバンク |
| | | 264 | マスタ・プログラム・カウンタ |
| | | 266 | スレーブ・プログラム・カウンタ |
| | | 300 | 命令メモリ・キャッシュ |
| | | 310 | 命令プロセッサ回路 |

【図1】

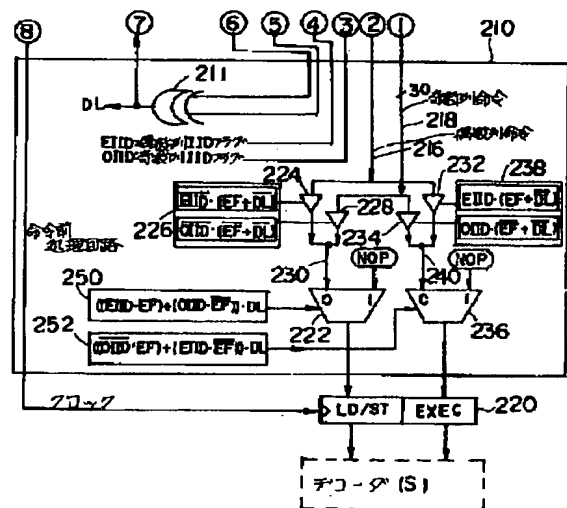
【図2】



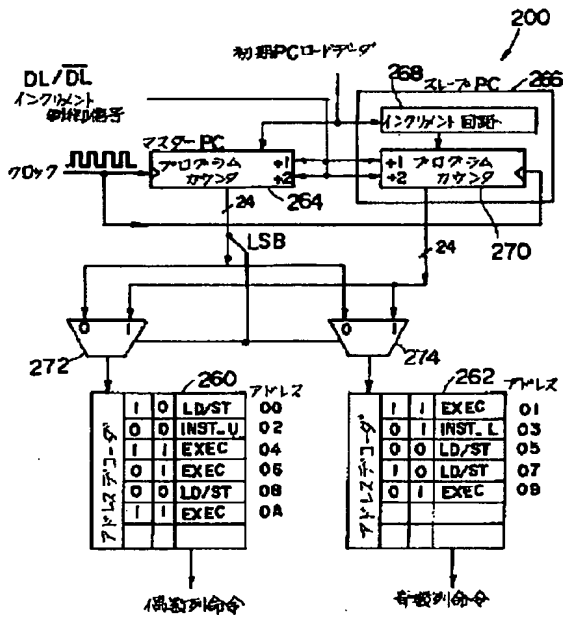
【図4】



【图 6】



【図7】



【図9】

